CLAIMS

What is claimed is:

1. A method for use in a multiple user computing environment logon user interface, the method comprising:

creating a separate desktop thread for each user that is authenticated;

creating a separate desktop associated with each desktop thread; and

maintaining a list of desktop threads that are created.

2. The method as recited in Claim 1, further comprising:

establishing a separate user environment associated with each desktop.

3. The method as recited in Claim 1, further comprising:

launching a separate user shell associated with each desktop.

4. The method as recited in Claim 1, further comprising:

selectively switching from a first desktop to a second desktop without terminating a desktop thread associated with the first desktop.

5. The method as recited in Claim 1, further comprising:

automatically switching from a first desktop to a second desktop without terminating a desktop thread associated with the first desktop launching a separate user shell associated with each desktop.

6. The method as recited in claim 5, wherein automatically switching from a first desktop to a second desktop occurs following a defined period of user inactivity.

7. The method as recited in Claim 1, further comprising:

selectively removing a desktop thread from the list of desktop threads when a user logs off.

8. A computer-readable medium having computer-executable instructions for performing steps comprising:

creating a separate desktop thread for each user that is authenticated;

creating a separate desktop associated with each desktop thread; and

maintaining a list of desktop threads that are created.

9. The computer-readable medium as recited in Claim 8, having further computer-executable instructions for performing the step of:

establishing a separate user environment associated with each desktop.

10. The computer-readable medium as recited in Claim 8, having further computer-executable instructions for performing the step of:

launching a separate user shell associated with each desktop.

11. The computer-readable medium as recited in Claim 8, having further computer-executable instructions for performing the step of:

selectively switching from a first desktop to a second desktop without terminating a desktop thread associated with the first desktop.

12. The computer-readable medium as recited in Claim 8, having further computer-executable instructions for performing the step of:

automatically switching from a first desktop to a second desktop without terminating a desktop thread associated with the first desktop launching a separate user shell associated with each desktop.

13. The computer-readable medium as recited in claim 12, wherein automatically switching from a first desktop to a second desktop occurs following a defined period of user inactivity.

14. The computer-readable medium as recited in Claim 8, having further computer-executable instructions for performing the step of:

selectively removing a desktop thread from the list of desktop threads when a user logs off.

15. An arrangement comprising:

memory having at least a portion of an operating system stored therein;

a processor operatively coupled to the memory and responsive to the operating system to create a separate desktop thread for each user that is authenticated during a logon process, create a separate desktop associated with each desktop thread, and maintain a list of desktop threads that are created.

16. The arrangement as recited in Claim 15, wherein the processor is further responsive to the operating system by establishing a separate user environment associated with each desktop.

17. The arrangement as recited in Claim 15, wherein the processor is further responsive to the operating system by launching a separate user shell associated with each desktop.

18. The arrangement as recited in Claim 15, wherein the processor is further responsive to the operating system by selectively switching from a first desktop to a second desktop without terminating a desktop thread associated with the first desktop.

19. The arrangement as recited in Claim 15, wherein the processor is further responsive to the operating system by automatically switching from a first desktop to a second desktop without terminating a desktop thread associated with the first desktop launching a separate user shell associated with each desktop.

20. The arrangement as recited in claim 19, wherein automatically switching from a first desktop to a second desktop occurs following a defined period of user inactivity.

21. The arrangement as recited in Claim 15, wherein the processor is further responsive to the operating system by selectively removing a desktop thread from the list of desktop threads when a user logs off.

22. A method for use in a multiple user computing environment logon user interface, the method comprising:

creating a separate remote process thread for each user that is authenticated;

creating a separate remote process associated with each remote process thread; and

maintaining a list of remote process threads that are created.

23. The method as recited in Claim 22, further comprising:

establishing a separate user environment associated with each remote process.

24. The method as recited in Claim 22, further comprising:

launching a separate user shell associated with each remote process.

25. The method as recited in Claim 22, further comprising:

selectively switching from a first remote process to a second remote process without terminating a remote process thread associated with the first remote process.

26. The method as recited in Claim 22, further comprising:

automatically switching from a first remote process to a second remote process without terminating a remote process thread associated with the first remote process launching a separate user shell associated with each remote process.

27. The method as recited in claim 26, wherein automatically switching from a first remote process to a second remote process occurs following a defined period of user inactivity.

28. The method as recited in Claim 22, further comprising:

selectively removing a remote process thread from the list of remote process threads when a user logs off.

29. A computer-readable medium having computer-executable instructions for performing steps comprising:

creating a separate remote process thread for each user that is authenticated;

creating a separate remote process associated with each remote process thread; and

maintaining a list of remote process threads that are created.

30. The computer-readable medium as recited in Claim 29, having further computer-executable instructions for performing the step of:

establishing a separate user environment associated with each remote process.

31. The computer-readable medium as recited in Claim 29, having further computer-executable instructions for performing the step of:

launching a separate user shell associated with each remote process.

32. The computer-readable medium as recited in Claim 29, having further computer-executable instructions for performing the step of:

selectively switching from a first remote process to a second remote process without terminating a remote process thread associated with the first remote process.

33. The computer-readable medium as recited in Claim 29, having further computer-executable instructions for performing the step of:

automatically switching from a first remote process to a second remote process without terminating a remote process thread associated with the first remote process launching a separate user shell associated with each remote process.

34. The computer-readable medium as recited in claim 33, wherein automatically switching from a first remote process to a second remote process occurs following a defined period of user inactivity.

35. The computer-readable medium as recited in Claim 29, having further computer-executable instructions for performing the step of:

selectively removing a remote process thread from the list of remote process threads when a user logs off.

36. An arrangement comprising:

memory having at least a portion of an operating system stored therein;

a processor operatively coupled to the memory and responsive to the operating system to create a separate remote process thread for each user that is authenticated during a logon process, create a separate remote process associated with each remote process thread, and maintain a list of remote process threads that are created.

37. The arrangement as recited in Claim 36, wherein the processor is further responsive to the operating system by establishing a separate user environment associated with each remote process.

38. The arrangement as recited in Claim 36, wherein the processor is further responsive to the operating system by launching a separate user shell associated with each remote process.

39. The arrangement as recited in Claim 36, wherein the processor is further responsive to the operating system by selectively switching from a first remote process to a second remote process without terminating a remote process thread associated with the first remote process.

40. The arrangement as recited in Claim 36, wherein the processor is further responsive to the operating system by automatically switching from a first remote process to a second remote process without terminating a remote process thread associated with the first remote process launching a separate user shell associated with each remote process.

41. The arrangement as recited in claim 40, wherein automatically switching from a first remote process to a second remote process occurs following a defined period of user inactivity.

42. The arrangement as recited in Claim 36, wherein the processor is further responsive to the operating system by selectively removing a remote process thread from the list of remote process threads when a user logs off.